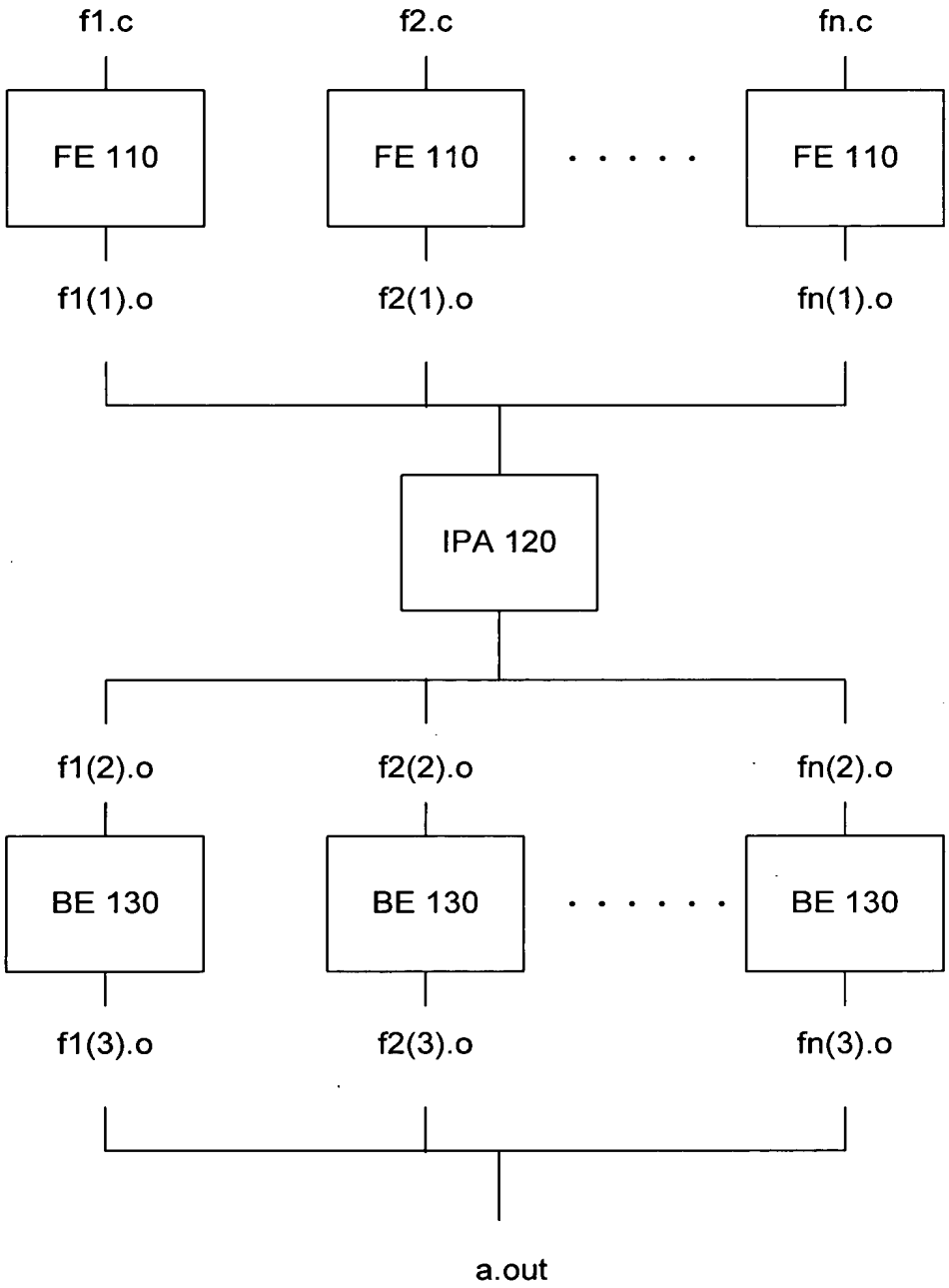
f1.c
f2.c
fn.c
FE 110
FE 110
FE 110
f1(1).o
f2(1).o
fn(1).o
IPA 120
f1(2).o
f2(2).o
fn(2).o
BE 130
BE 130
BE 130
f1(3).o
f2(3).o
fn(3).o
a.out

FIG. 1

100

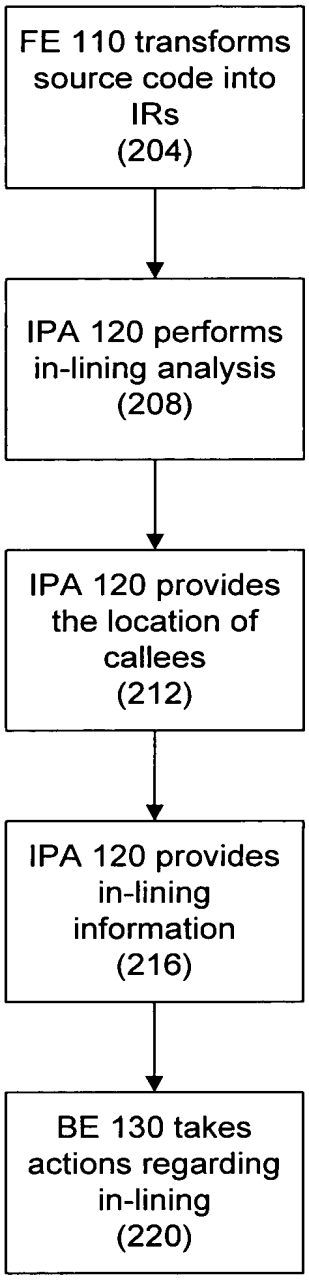
FE 110 transforms source code into IRs (204)
IPA 120 performs in-lining analysis (208)
IPA 120 provides the location of callees (212)
IPA 120 provides in-lining information (216)
BE 130 takes actions regarding in-lining (220)

FIG. 2

f1.c

```
305 foo (){
  310   bar()
}
```

f2.c

```
315 bar (){
  320   func()
}
```

f3.c

```
325 func (){
}
```

FIG. 3A

f1(1).o

```
305 foo (){
  310   bar()
}
```

f2(1).o

```
315 bar (){
  320   func()
}
```

f3(1).o

```
325 func (){
}
```

FIG. 3B

f1(2).o

```
305 foo (){
  310   bar()
}
330 bar(){
}
```

f2(2).o

```
315 bar (){
  320   func()
}
```

f3(2).o

```
325 func (){
}
```

FIG. 3C

f1(3).o

```
305 foo (){
  310   function
bar() in-lined herein
}
330 bar(){
being deleted after
in-lined into
function foo()
}
```

f2(3).o

```
315 bar (){
  being deleted
}
```

f3(3).o

```
325 func (){
}
```

FIG. 3D

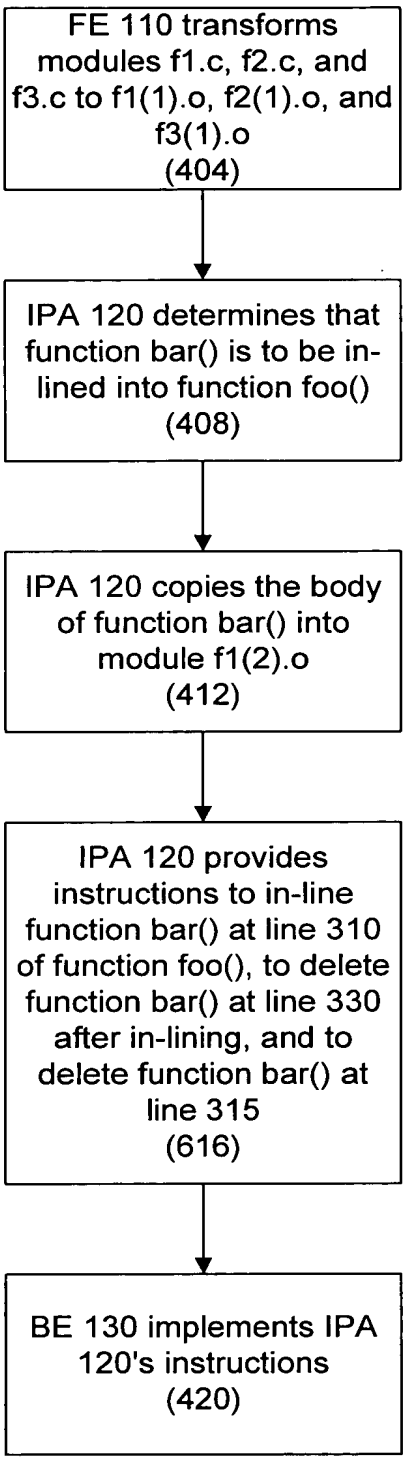
FE 110 transforms modules f1.c, f2.c, and f3.c to f1(1).o, f2(1).o, and f3(1).o
(404)
IPA 120 determines that function bar() is to be in-lined into function foo()
(408)
IPA 120 copies the body of function bar() into module f1(2).o
(412)
IPA 120 provides instructions to in-line function bar() at line 310 of function foo(), to delete function bar() at line 330 after in-lining, and to delete function bar() at line 315
(616)
BE 130 implements IPA 120's instructions
(420)

FIG. 4

ff1.c

```
505 ffoo (){
  510   bbar()
}
```

ff2.c

```
515 bbar (){
  520   ffunc()
}
```

ff3.c

```
525 ffunc (){
}
```

FIG. 5A

ff1(1).o

```
505 ffoo (){
  510   bbar()
}
```

ff2(1).o

```
515 bbar (){
  520   ffunc()
}
```

ff3(1).o

```
525 ffunc (){
}
```

FIG. 5B

ff1(2).o

```
505 ffoo (){
  510   bbar()
}
530 bbar(){
}
540 ffunc(){
}
```

ff2(2).o

```
515 bbar (){
  520   ffunc()
}
```

ff3(2).o

```
525 ffunc (){
}
```

FIG. 5C

ff1(3).o

```
505 ffoo (){
  510   function
bbar() and ffunc()
being in-lined herein
}
530 bbar(){
being deleted after
in-lining
}
540 ffunc(){
being deleted after
in-lining
}
```

ff2(3).o

```
515 bbar (){
  being deleted
}
```

ff3(3).o

```
525 ffunc (){
  being deleted
}
```

FIG. 5D

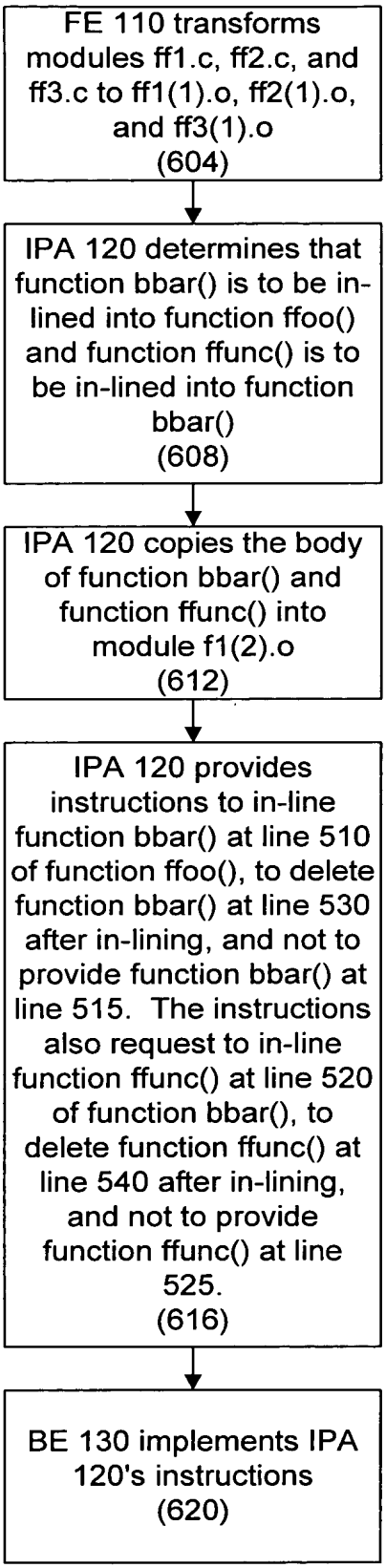
FE 110 transforms modules ff1.c, ff2.c, and ff3.c to ff1(1).o, ff2(1).o, and ff3(1).o
(604)
IPA 120 determines that function bbar() is to be in-lined into function ffoo() and function ffunc() is to be in-lined into function bbar()
(608)
IPA 120 copies the body of function bbar() and function ffunc() into module f1(2).o
(612)
IPA 120 provides instructions to in-line function bbar() at line 510 of function ffoo(), to delete function bbar() at line 530 after in-lining, and not to provide function bbar() at line 515. The instructions also request to in-line function ffunc() at line 520 of function bbar(), to delete function ffunc() at line 540 after in-lining, and not to provide function ffunc() at line 525.
(616)
BE 130 implements IPA 120's instructions
(620)

FIG. 6

```
bar(int p) {

    if (p==0) {

    710
    . . .
    720

    } else {

    730
    . . .
    740
    }

}
```

FIG. 7A

```
bar_clone_1(int p) {

    if (p==0) {

    710
    . . .
    720

    }

}
```

FIG. 7B

```
bar_clone_2(int p) {

    if (p!=0) {

    730
    . . .
    740

    }

}
```

FIG. 7C

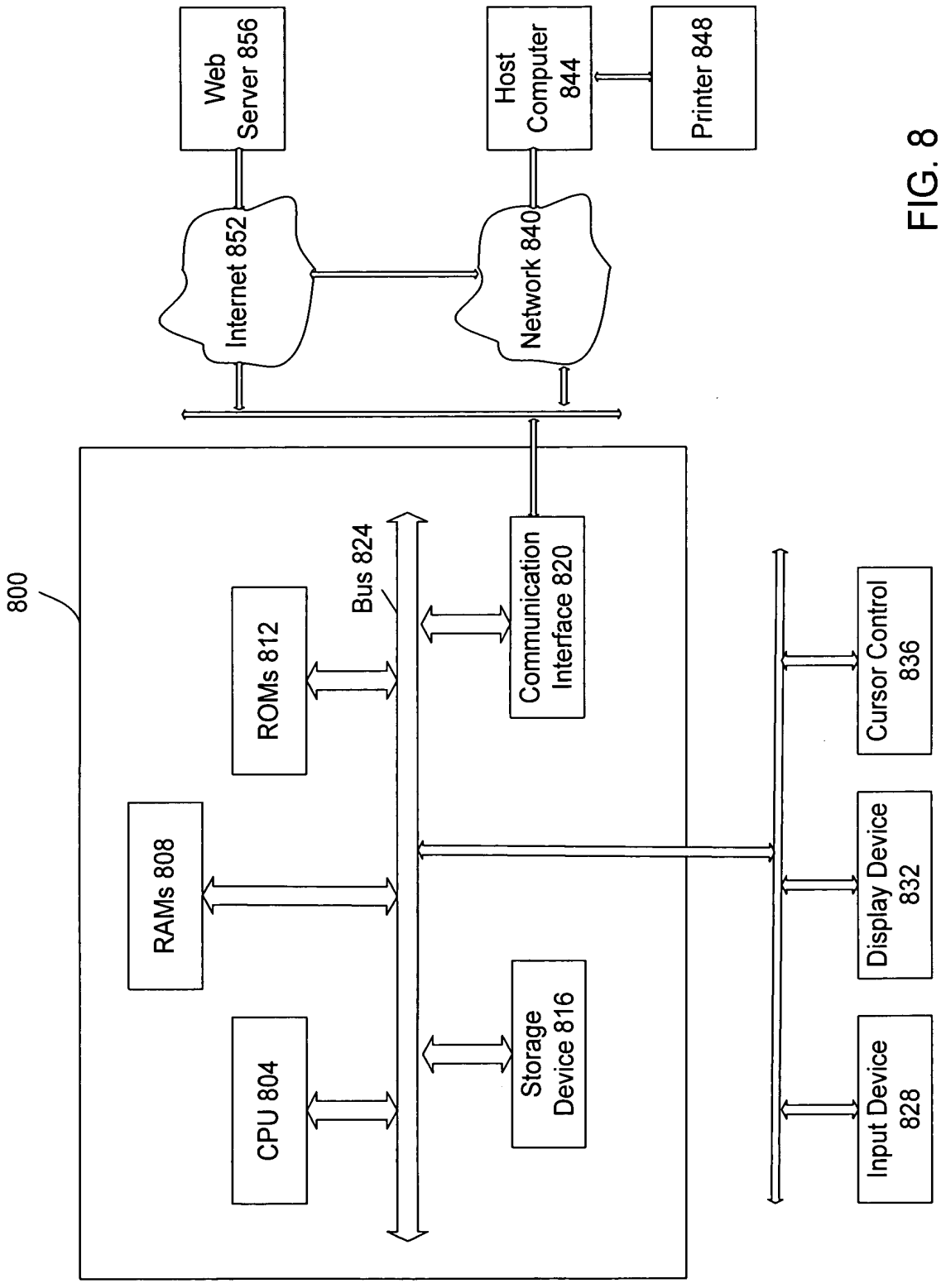
Web Server 856
Host Computer 844
Printer 848
Internet 852
Network 840
800
Bus 824
Communication Interface 820
ROMs 812
RAMs 808
CPU 804
Storage Device 816
Cursor Control 836
Display Device 832
Input Device 828

FIG. 8